

Towards JPEG-Compression Invariance for Adversarial Optimization

Amon Soares de Souza¹^a, Andreas Meißner²^b and Michaela Geierhos¹^c

¹University of the Bundeswehr Munich, Research Institute CODE, Werner-Heisenberg-Weg 39, 85577 Neubiberg, Germany

²ZITiS, Big Data, Zamdorfer Str. 88, 81677 Munich, Germany

Keywords: Adversarial Optimization, Adversarial Attacks, Image Classification.

Abstract: Adversarial image processing attacks aim to strike a fine balance between pattern visibility and target model error. This balance ideally results in a sample that maintains high visual fidelity to the original image, but forces the model to output the target of the attack, and is therefore particularly susceptible to transformations by post-processing such as compression. JPEG compression, which is inherently non-differentiable and an integral part of almost every web application, therefore severely limits the set of possible use cases for attacks. Although differentiable JPEG approximations have been proposed, they (1) have not been extended to the stronger and less perceptible optimization-based attacks, and (2) have been insufficiently evaluated. Constrained adversarial optimization allows for a strong combination of success rate and high visual fidelity to the original sample. We present a novel robust attack based on constrained optimization and an adaptive compression search. We show that our attack outperforms current robust methods for gradient projection attacks for the same amount of applied perturbation, suggesting a more effective trade-off between perturbation and attack success rate. The code is available here: <https://github.com/amonsoes/frew>.

1 INTRODUCTION

Adversarial attacks provide a straightforward way to improve and evaluate the robustness of deep learning models. Methods that project the input based on the sign of the gradient of a surrogate model are commonly used to improve model robustness because they are less computationally intensive and can be used in the inner loop of adversarial training (Madry et al., 2018). In contrast, attacks based on adversarial optimization assess model robustness by solving a computationally expensive constrained optimization problem that generates adversarial samples that closely resemble the original image while fooling the target model (Szegedy et al., 2014).

Optimally, the adversarial sample is the sample closest to the original image (according to some distortion measure) that forces the model to output the target (Szegedy et al., 2014). This fine balance is easily disrupted by transformations that change pixels or groups of pixels, such as compression. JPEG compression is an integral part of almost every application that processes and stores images or other data,

severely limiting the use cases for attacks. Since this type of compression is inherently non-differentiable, it cannot easily be used in an optimization scheme (Shin and Song, 2017). While there have been successful attempts to incorporate a differentiable approximation into gradient projection-based attacks, these works have not attempted to do the same for optimization-based attacks, which are often less noticeable and harder to defend against.





(a) Original


(b) RCW

Figure 1: Comparison of the adversarial samples generated by RCW with the original sample. Zooming in, you can see that high frequency details have been removed.

Our RCW attack builds on Carlini and Wagner (2017). Current approaches mainly rely on a gradient ensemble over a set of quality settings. However, gradient ensembles would introduce an additional inner

^a <https://orcid.org/0009-0000-7978-1281>

^b <https://orcid.org/0000-0002-6200-7553>

^c <https://orcid.org/0000-0002-8180-5606>

loop into the adversarial optimization, and resulting in undesirably long computation times. Instead of using gradient ensemble methods, this attack performs a search for the JPEG quality factor by querying the target system once. This produces a pair $(\mathbf{x}, \mathbf{x}')$, where \mathbf{x}' is the compressed output of the target system. We use this pair to perform a search for the quality setting used by minimizing the L_2 distance from \mathbf{x}' to $JPEG(\mathbf{x}, q)$, where $JPEG$ is our JPEG algorithm and q is the quality setting. This search eliminates the need to query every possible quality setting to perform compression, and finds the optimal quality setting in a fraction of the steps compared to a brute-force approach. By incorporating the differentiable JPEG approximation into constrained adversarial optimization, we show that adversarial attacks do not require a high order of perturbation magnitude to overcome compression. Adversarial samples generated by RCW retain high visual fidelity and are still effective (see Figure 1). For further comparisons between RCW-generated adversarial samples and their respective original images, see Figure 3. To summarize our contributions in this paper:

1. We introduce a differentiable JPEG approximation for optimization-based attacks, which has only been applied to gradient projection-based attacks (Shi et al., 2021; Reich et al., 2024).
2. We propose an alternative to the gradient ensemble methods found in the current approaches (Shin and Song, 2017; Reich et al., 2024) in order to successfully induce robustness against JPEG compression with varying compression settings for adversarial optimization.
3. In addition to white-box and black-box evaluations and benchmarks on target models hardened by adversarial training, we compare the *perceived* distortion of our samples with those of the related work. These experiments have not yet been addressed by the related work.
4. We show that our adversarial samples can overcome compression while maintaining high image fidelity, and report the differences in success rate and average distortion compared to the current state of the art. Our experiments indicate that our attack results in a better balance between attack success rate and applied distortion.
5. We extensively analyze our compression adaptation search procedure and perform an ablation study that highlights the benefits of extending optimization-based attacks to include the JPEG approximation in the loss function as well as the compression setting search for varying compression rates.

2 RELATED WORK

There is a rich body of work on adversarial attacks, covering a variety of approaches and use cases.

Szegedy et al. (2014) introduced adversarial samples by performing constrained optimization on the input using an adversarial loss. Optimization-based attacks require a computationally expensive process, but are usually effective because (1) it is impractical to use optimization-based attacks in adversarial training, and (2) they usually result in an optimum where the attack fools the model with a minimum required distortion (Carlini and Wagner, 2017).

Gradient projection methods work very differently. As their name implies, these methods project the input in the direction of the sign of the gradient to increase the loss of the model. They are often used to perform adversarial training (Goodfellow et al., 2015; Wang and He, 2021). As far as distortion is concerned, these latter methods are usually L_∞ bounded, which means that these attacks often result in perturbations where most pixels are changed to their maximum extent. Optimization-based attacks often use the L_2 norm as a constraint, resulting in a distortion that is not maximized for every pixel (Goodfellow et al., 2015; Carlini and Wagner, 2017; Wang and He, 2021). In terms of use cases, both approaches can be used as the basis for targeted and untargeted attacks, in both white box and black box environments.

While there have been considerations that address undesirable characteristics of these attacks, such as attack visibility, the lack of smoothness (Luo et al., 2022), and the challenges of deploying attacks in the physical world (Kurakin et al., 2017), most attacks only consider settings in the uncompressed domain. This is surprising, given that JPEG compression can easily suppress the adversarial noise of most attacks, and is even considered to function as an adversarial defense by various defense methods (Liu et al., 2019).

Shi et al. (2021) successfully produce adversarial images resistant to JPEG compression by introducing a procedure called *adversarial rounding*. Instead of distorting pixel values, this method makes adjustments in the patched discrete cosine transform (DCT) projection of an initial adversarial sample produced by FGSM (Goodfellow et al., 2015) and BIM (Kurakin et al., 2017). They distinguish between *fast adversarial rounding* and *iterative adversarial rounding*. The first method produces an adversarial DCT projection by quantizing the DCT patches in the direction of the gradient to increase the model loss. This approach also prioritizes DCT components that have a greater impact on the model decision (Shi et al., 2021).

Shin and Song (2017) propose a method to include a differentiable JPEG approximation in projection-based attacks, specifically to target models that use JPEG as a defense. They argue that JPEG, being a lossy compression method, results in an image that preserves semantic details but discards the adversarial perturbations, making the attack less effective. Quantization in JPEG involves rounding the coefficients obtained by the DCT transform to the nearest integer. This produces gradients that are everywhere 0, making the function non-differentiable. They design an approximation that adds the cubed difference between the original coefficient and the rounded coefficient during quantization. They extend FGSM (Goodfellow et al., 2015) and BIM (Kurakin et al., 2017) with their JPEG approximation, allowing them to incorporate compression into the gradient computation. However, they only extend attacks based on gradient projection and omit optimization-based attacks (Shin and Song, 2017). Improving on the work of Shin and Song (2017), Reich et al. (2024) also include a differentiable JPEG approximation in projection-based attacks, but they extend the surrogate approach by remodeling the computations to obtain the quantization table.

Other work suggests that the reliability of attacks can be inherently improved by considering additional characteristics of adversarial attacks. Zhao et al. (2020) propose to create adversarial examples by perturbing images with respect to the perceptual color distance (PerC). They argue that color distances are less perceptible because color perceived by the human visual system (HVS) does not change uniformly with distance in the RGB space. Instead of using a traditional L_p norm as a constraint during optimization, they use the CIEDE2000 color metric. They also introduce an alternating optimization procedure called PerC-AL, which computes the adversarial loss for backpropagation when the sample is not adversarial, and the image quality loss with CIEDE2000 when the sample is adversarial (Zhao et al., 2020).

3 METHOD

In the following section, we define the threat model in which we conduct our attack to bypass the target system’s JPEG compression. After outlining the procedure, we will examine the characteristics of the RCW attack. We use the standard definition of adversarial samples, where δ is the perturbation, \mathbf{x} is the original input, $y \in \mathbf{Y}$ is the ground truth, ϵ is the maximum perturbation threshold, f is the target model and θ_f are its parameters. A sample is adversarial if the following

holds (Szegedy et al., 2014; Goodfellow et al., 2015; Kurakin et al., 2017; Shin and Song, 2017; Zhao et al., 2020; Wang and He, 2021; Luo et al., 2022):

1. $\mathbf{x} + \delta \in [0, 1]$
2. $f(\mathbf{x} + \delta; \theta_f) = \hat{y}; \hat{y} \in \mathbf{Y} \setminus y$
3. $\forall \delta \in \delta: |\delta| \leq \epsilon$

In the following, $\mathbf{x} + \delta$ equals \mathbf{x}_{adv} . We define a threat model, outline the attack procedure, and propose the robust Carlini and Wagner attack method (RCW).

3.1 Threat Model

Akhtar et al. (2021) define a threat model as the adversarial conditions against which a defense mechanism is tested to verify its effectiveness. We adapt this concept and define *threat model* as an interaction between an adversary and a target system. In this interaction, the adversary tries to force the target model, which is part of the target system’s environment, to produce false output. In all of our diagrams (see Figure 2 and Figure 4), the red elements are features of the adversary, while the blue elements are features of the target system. Both terms are defined below.

3.1.1 Target System

Our approach requires that a target system includes at least a target JPEG compression algorithm J_{target} that compresses the input \mathbf{x} , and a target model ϕ that processes the compressed input to produce the desired output. As a minimal working example, our target system can be defined as

$$T(J_{target}, q_{target}, \phi, \mathbf{x}) = \phi(J_{target}(\mathbf{x}, q_{target})) \quad (1)$$

In real-world applications, such a target system is often found in social media, where user-uploaded images are compressed and then processed by a model that performs some desired task.

3.1.2 Adversary

Akhtar et al. (2021) define an adversary as the agent (i.e., the attacker) who creates an adversarial example. Based on this definition, we define our adversary as follows. Let *Attack* be an adversarial attack and \mathbf{x} be the input. The output of *Attack* is an adversarial sample \mathbf{x}_{adv} computed using a surrogate model $\hat{\phi}$.

$$A(\text{Attack}, \mathbf{x}, \hat{\phi}) = \text{Attack}(\mathbf{x}; \hat{\phi}) \quad (2)$$

In our scenario, the adversary can also query the target’s JPEG algorithm J_{target} to compress the input \mathbf{x} .

3.2 Attack Procedure

For attacks performed with our method, we provide a complete outline of the workflow in Figure 2. First, we query the target system’s JPEG compression $J_{target}(\mathbf{x}, q_{target})$ with \mathbf{x} to obtain the compressed counterpart \mathbf{x}'_g . Both are passed as a tuple $(\mathbf{x}, \mathbf{x}'_g)$ to a procedure called compression adaptation search, which returns the quality factor q^* that best mimics the compression setting q_{target} . This quality factor is then passed to RCW, our *Attack*, to compute the robust adversarial sample \mathbf{x}_{adv} .

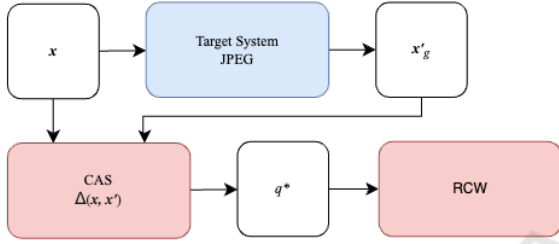


Figure 2: Graphical representation of the RCW attack flow. The attack requires a query to the target system’s JPEG algorithm. It then performs CAS to find the best compression setting q^* .

3.2.1 Compression Adaptation Search (CAS)

The output of the query is used to perform a line search that minimizes the L_2 distance from \mathbf{x}'_g to $J(\mathbf{x}, q)$, where J is our JPEG algorithm and q is the quality setting. The goal of this search, which we call compression adaptation search (CAS), is to find the quality setting q^* that best mimics the quality setting q_{target} of the target system’s JPEG algorithm J_t . Let Δ be the distance L_2 . Let \mathbf{x} be the uncompressed image and \mathbf{x}'_g be the compressed target image, which is the output of the JPEG compression algorithm of the target system J_{target} . CAS has several parameters to control the search. Let p be the direction of the line search (e.g., whether the value of q is ascending or descending). s_t is the step size, decreasing continuously by τ . It is used to scale the step size, which is given by the distance $d_t = \Delta(J(\mathbf{x}, q_t), \mathbf{x}'_g)$. Let q_t be the current quality setting, randomly initialized with an integer in the range $\{1, 99\}$ in q_0 . In a few cases an intermediate q_t resulted in a higher distance d_{t+1} , even though the search was approaching q^* in the right direction. Therefore we allow for 2 exploration steps (denoted as β) before changing the search direction in case d_{t+1} does not improve on d_t . Finally, let γ be an early termination criterion that stops the search if d_t does not improve for ten steps. The whole procedure is given in Algorithm 1.

```

Input:  $\mathbf{x}$ ; // uncompressed image
Input:  $\mathbf{x}'_g$ ; // compressed target image
Input:  $d_g$ ; // target distance
Result:  $q^*$ ; // best quality setting
 $p \leftarrow -1$ ; // search direction
 $\tau \leftarrow 0.99$ ; // temperature
 $s_0 \leftarrow 1.0$ ; // schedule
 $d_0 \leftarrow 1e10$ ; // init best distance
 $d^* \leftarrow d_0$ ; // best distance
 $q_0 \leftarrow r(1, 99)$ ; // random init of  $q$ 
 $q^* \leftarrow q$ ; // best  $q$ 
 $\gamma \leftarrow 0$ ; // early termination criterion
 $\beta \leftarrow 0$ ; // exploration criterion
while  $d^* > d_g$  do
   $\mathbf{x}' \leftarrow J(\mathbf{x}, q_t)$ ;
   $d_{t+1} \leftarrow \Delta(\mathbf{x}', \mathbf{x}'_g)$ ;
  if  $d_{t+1} \geq d_t$  then
     $\gamma \leftarrow \gamma + 1$ ;
     $\beta \leftarrow \beta + 1$ ;
    if  $\beta \geq 2$  then
       $p \leftarrow p \cdot -1$ ;
       $\beta \leftarrow 0$ ;
    end
  end
  else if  $d_{t+1} < d_t$  then
     $\gamma \leftarrow 0$ ;
     $\beta \leftarrow 0$ ;
    if  $d_{t+1} < d^*$  then
       $d^* \leftarrow d_{t+1}$ ;
       $q^* \leftarrow q_t$ ;
    end
  end
  if  $\gamma > 10$  then
    /* quit search early */
    return  $q^*$ ;
  end
   $s_{t+1} \leftarrow s_t \cdot \tau$ ;
   $q_{t+1} \leftarrow \min(\max(q_t + p \cdot (s_{t+1} \cdot d_{t+1}), 1), 99)$ ;
end
return  $q^*$ 
  
```

Algorithm 1: Compression Adaptation Search (CAS).

3.2.2 RCW

Based on adversarial optimization, our attack uses a differentiable approximation of JPEG along with the output q^* of CAS to compute the robust adversarial sample \mathbf{x}_{adv} .

Differentiable JPEG. JPEG compression is inherently difficult to use with gradient descent. This is due to some internal computations that are not differentiable. There are four steps in JPEG encoding: (1)

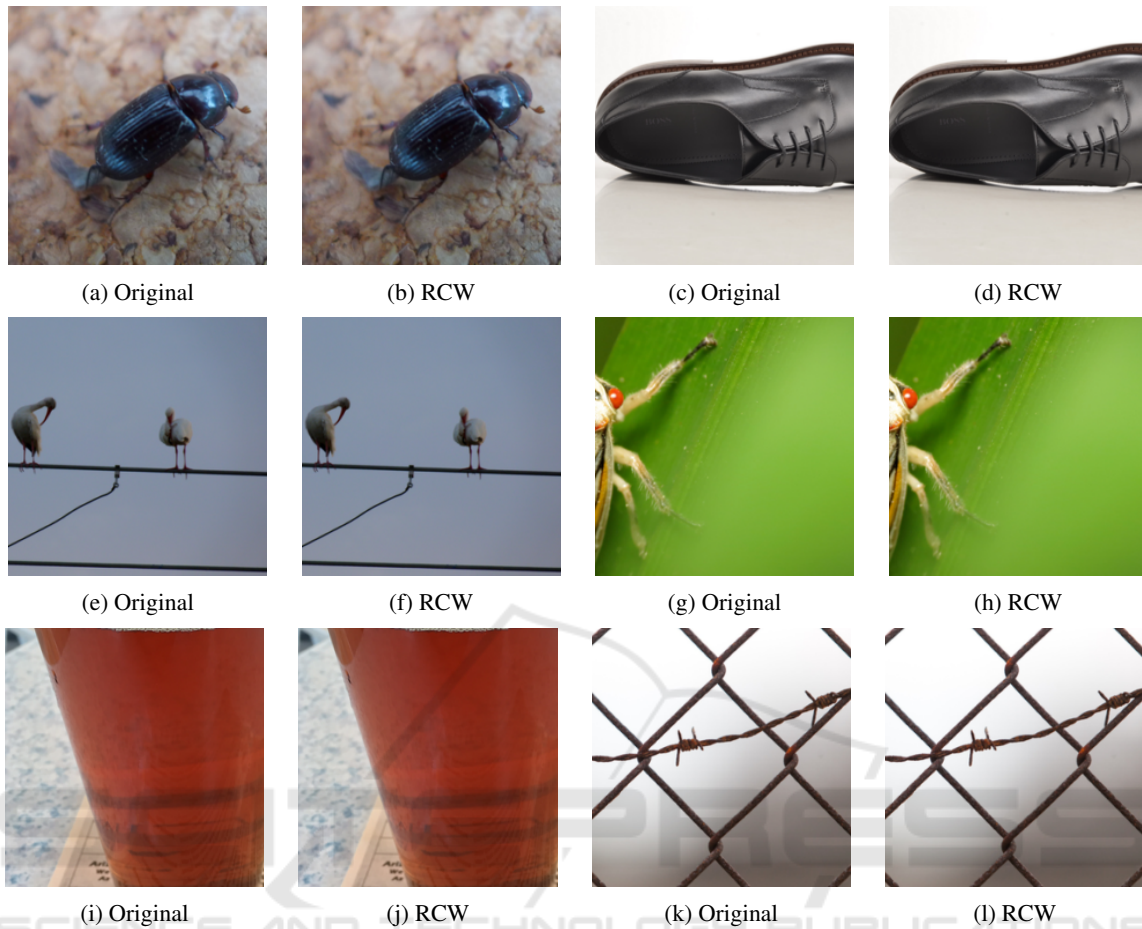


Figure 3: Comparison of the adversarial samples produced by RCW to the original sample.

color conversion, where the RGB is mapped to the YcbCr color space (2) *chroma subsampling*, where the two chroma channels, Cb and Cr, are downsampled by a factor (3) *patched DCT*, which usually first divides the input into 8x8 patches and then calculates the DCT for each patch, and (4) *quantization*, which maps the output of the DCT to an integer by a quantization table that is predefined by the chosen JPEG quality (Shin and Song, 2017; Reich et al., 2024). The fourth step, quantization, relies on rounding and floor functions, resulting in gradients that are almost always zero. Shin and Song (2017) proposed a polynomial approximation of the rounding function $\lfloor \mathbf{x} \rfloor_{approx} = \lfloor \mathbf{x} \rfloor + (\mathbf{x} + \lfloor \mathbf{x} \rfloor)^3$ and they additionally reformulate the scaling of the quantization table by the JPEG quality. Other methods approximate the non-differentiable function of the compression by using a straight-through estimator. This method uses the true, non-differentiable method for the forward pass and a constant gradient of one in the backward pass (Reich et al., 2024). For our purposes, we use the surrogate model approach outlined in Reich et al. (2024), which

extends the existing surrogate approach of Shin and Song (2017) by remodeling the computations to obtain the quantization table.

Adversarial Optimization. Adding a compression approximation term to the adversarial optimization can yield stronger, more reliable targeted adversarial samples that maintain high visual fidelity to the original sample. Based on Carlini and Wagner (2017), we adapt their method to compute the adversarial loss by extending the loss computation to include compression in the backward pass. The adversarial loss function f measures the effectiveness of the adversarial sample. Let t be the index of the target label, q the compression quality, \mathbf{x}_{adv} the adversarial sample, κ the confidence factor (which increases the probability of success for additional distortion), and J_d the differentiable JPEG compression in Equation 1. Furthermore, let Z be a mapping of an input to a set of logits, where each logit represents a class. The underlying parameters of Z are provided by the surrogate model

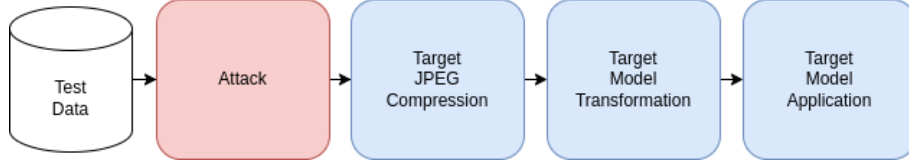


Figure 4: Graphical representation of our experiment pipeline.

$\hat{\phi}$. The confidence factor κ controls the desired effectiveness of the adversarial sample, with higher values of κ requiring a more effective adversarial perturbation.

$$f(\mathbf{x}_{adv}, y_t; q) = \max\{\max[Z(J_d(\mathbf{x}_{adv}); q)_i : i \neq t] - Z(J_d(\mathbf{x}_{adv}); q)_t, -\kappa\} \quad (3)$$

An appropriate full-reference image quality metric imposes the constraint. Let χ be an appropriate full-reference image quality metric that evaluates the original sample \mathbf{x} and its adversarial counterpart \mathbf{x}_{adv} , where χ measures the visual fidelity of \mathbf{x}_{adv} to \mathbf{x} . Let c be a trade-off constant that balances the adversarial loss f with the image quality loss. Our complete loss function can be defined as:

$$\Psi(\mathbf{x}, \mathbf{x}_{adv}, y_t, q) = \chi(\mathbf{x}, \mathbf{x}_{adv}) + c \cdot f(\mathbf{x}_{adv}, y_t; q) \quad (4)$$

Accounting for Varying Compression Magnitudes.

This sets up the constrained optimization problem for finding an appropriate adversarial sample using RCW (see Figure 2). However, in the current design, we would have to guess the correct quality setting q to use in Equation 2.

Current attacks account for different JPEG compression rates by using a gradient ensemble over a set of compression values (Shin and Song, 2017; Reich et al., 2024). Using this approach in adversarial optimization would require an additional inner loop for the gradient ensemble computation and would drastically increase the computation time, as the attack would require $n \times m$ successive forward- and backward calls (instead of n) to the surrogate model $\hat{\phi}$ to compute the adversarial sample, where n is the number of steps and m is the set of compression settings for the gradient ensemble method.

Therefore, the correct quality setting q^* is first computed by CAS (see Section 3.2.1), RCW minimizes the adversarial loss by Equation 2, using the estimate q^* as the compression setting. The adversarial optimization problem can be defined as follows. Let δ be the perturbation that is added to \mathbf{x} to obtain \mathbf{x}_{adv} .

$$\min_{\delta} \Psi(\mathbf{x}, \mathbf{x}_{adv}, y_t, q^*) \quad (5)$$

4 EXPERIMENTS

To perform well at all compression settings, reliable attacks are needed. Therefore, we perform all tests on every $q \in \{70, 80, 90\}$. This range is usually considered for current work using compression (Cozzolino et al., 2023). For a fair comparison with the state of the art, we report the success rate for each q using the same amount of distortion (expressed by \bar{D}). If the distortion varies in between compression settings, we report the average distortion of all compression settings. Due to different underlying mechanisms, not all attacks share the same set of hyperparameters. We only perform targeted attacks, where the target is the most likely label next to the ground truth. This is similar to the untargeted attacks. Our surrogate model $\hat{\phi}$ is a ResNet (He et al., 2016) pre-trained on the respective test dataset. We will consider three scenarios: (1) white box ($\hat{\phi} = \phi$), (2) black box ($\hat{\phi} \neq \phi$), and (3) white box models where the model has been hardened by adversarial training. Our results can be found in the corresponding Table 1, Table 2, and Table 3.

4.1 Pipeline

For a realistic scenario, we design our experiment pipeline as follows. (1) **Test Data:** We load the data and apply basic transformations such as center-cropping and resizing. (2) **Attack:** We apply the attack to the image and project the result to the original $[0, 1]$ range. (3) **Target JPEG Compression:** To simulate typical behavior in web applications, we now apply the JPEG compression. (4) **Target Model Transformation:** We apply the transformations required by the target model. (5) **Target Model Application:** The compressed and transformed adversarial sample is applied to the target model. Figure 4 illustrates the process.

4.2 Settings

We compare our RCW method with three state-of-the-art approaches: Two iterative attacks with different JPEG approximations (Reich et al., 2024; Shin and Song, 2017), called JpegIFGSM, and Fast Adver-

Table 1: Conditional average distortion \bar{D} and attack success rate (ASR) per compression setting q in a white box scenario.

White Box				
Attack	CAD \bar{D}	ASR q=70	ASR q=80	ASR q=90
JpegIFGSM (Reich et al., 2024)	0.1340	0.193	0.328	0.343
JpegIFGSM (Shin and Song, 2017)	0.1330	0.178	0.308	0.338
FAR (Shi et al., 2021)	0.1218	0.019	0.018	0.023
RCW (ours)	0.1210	0.642	0.662	0.663

Table 2: Conditional average distortion \bar{D} and attack success rate (ASR) per compression setting q in a black box scenario.

Black Box				
Attack	CAD \bar{D}	ASR q=70	ASR q=80	ASR q=90
JpegBIM (Reich et al., 2024)	0.1331	0.067	0.063	0.049
JpegBIM (Shin and Song, 2017)	0.1320	0.061	0.060	0.045
FAR (Shi et al., 2021)	0.2306	0.081	0.081	0.078
RCW (ours)	0.0873	0.066	0.061	0.044

serial Rounding (FAR) (Shi et al., 2021). Our settings are chosen so that the amount of distortion caused by the attacks is roughly equal. For RCW, we set c to 0.5, the learning rate α to $1e-05$, and the number of optimization steps n to 10,000. When running CAS for RCW, we set the temperature τ to 0.99. For JpegIFGSM, we set the L_∞ perturbation bound to $\epsilon = 0.0004$, the number of steps to $n = 7$, and the step size to $\alpha = \frac{\epsilon}{n}$. For FAR, we use $\epsilon=9e-05$ for the base adversarial sample and set $\eta = 0.3$ to compute the percentile of the DCT components that are adjusted. JpegIFGSM (Shin and Song, 2017; Reich et al., 2024) accounts for different compression strengths by computing and ensembling the gradient over a set of N compression values. In our experiments, we set N to 6, which means that compression settings from 99 to 70, in decrements of 5, are used to compute the gradient. FAR (Shi et al., 2021) does not use any procedure to account for different compression rates, so we set $q = 80$ for all of its runs.

4.3 Test Data

All of our experiments use the NIPS 2017 adversarial competition dataset (Kurakin et al., 2018). This dataset consists of 1,000 images from the ImageNet-1K challenge, which contains a wide variety of image classes and presents a challenging and realistic problem. In addition to benchmarking against standard attacks, this dataset allows us to compare our method with related approaches. We do not evaluate on the CIFAR datasets, as some work (Tramèr et al., 2018) suggests that the methods tested on these datasets show poor generalization to more complicated tasks.

4.4 Evaluation Metrics

Our experimental results using the following metrics ASR and CAD can be found in Table 1, Table 2, and Table 3, while the results using the metrics MAD and DISTS are shown in Table 4.

4.4.1 Attack Success Rate (ASR)

The frequency with which an attack successfully causes the target network to misclassify inputs should be quantified in an appropriate metric. To accurately measure the performance of the attack, we define a subset \mathbf{X}_t of the original test dataset \mathbf{X} that contains data points that were initially correctly classified by the target network. Within this subset, the proportion of data points for which the attack caused a misclassification is called the attack success rate (ASR). Let t be the ground truth of a data point \mathbf{x} , ϕ the target network, N the number of data points in \mathbf{X}_t , and α the attack (Wang and He, 2021).

$$\mathbf{X}_t = \{\mathbf{x} \in \mathbf{X} | \phi(\mathbf{x}, \theta) = t\} \quad (6)$$

$$\mathbf{X}_t^{success} = \{\mathbf{x} \in \mathbf{X}_t | \phi(\alpha(\mathbf{x}), \theta) \neq t\} \quad (7)$$

$$ASR(\phi(\mathbf{X}_t, \theta), T) = \frac{|\mathbf{X}_t^{success}|}{N} \quad (8)$$

4.4.2 Conditional Average Distortion (CAD)

In addition to ASR, the conditional average distortion \bar{D} measures the average distance of an adversarial example $\hat{\mathbf{x}} = f(\mathbf{x})$ from the original data point \mathbf{x} , where $\mathbf{x} \in \mathbf{X}_t^{success}$. This distance is measured using the L2 norm, which was selected as the distortion metric.

$$\bar{D}(f, \mathbf{X}_t^{success}) = \frac{1}{|\mathbf{X}_t^{success}|} \sum_{\mathbf{x} \in \mathbf{X}_t^{success}} \|f(\mathbf{x}) - \mathbf{x}\|_2 \quad (9)$$

Since FAR (Shi et al., 2021) produces JPEG images, we compare the adversarial sample produced by FAR with the compressed version of the respective original image, compressed with the same quality setting as FAR uses internally. This way, only the distortion caused by the attack is measured, as intended.

4.4.3 Most Apparent Distortion (MAD)

Fezza et al. (2019) compared several full-reference image quality metrics and found that most apparent distortion (MAD) was most consistent with human perception. Based on this finding, we will use L_p norms, such as \bar{D} , exclusively as distortion measures, while using MAD and DISTS to estimate the perceived distortion of adversarial samples. MAD is a weighted linear combination of two components: the near-threshold distortion D_{near} , which captures early human vision, and the suprathreshold distortion D_{supra} , which captures more obvious distortions. Let α , β be the balancing scalars.

$$MAD(\mathbf{x}_{adv}, \mathbf{x}) = \alpha \cdot D_{near}(\mathbf{x}_{adv}, \mathbf{x}) + \beta \cdot D_{supra}(\mathbf{x}_{adv}, \mathbf{x}) \quad (10)$$

4.4.4 Deep Image Structure and Texture Similarity (DISTS)

In addition to MAD, we include a newer full-reference image quality evaluation method. Deep Image Structure and Texture Similarity (DISTS) (Ding et al., 2022) is a model-based quality score that performs well with human perceptual scores on traditional image quality evaluation databases. Unlike existing image quality scoring methods, DISTS provides good human quality scores for both textures and natural photographs (Ding et al., 2022). It scores an image based on the weighted linear combination of a structural similarity model S and a textual similarity model T . Let α , β be balancing scalars, l the number of layers in the networks, and w_i their corresponding weights.

$$DISTS(\mathbf{x}_{adv}, \mathbf{x}) = \sum_i^l w_i (\alpha \cdot S(\mathbf{x}_{adv}, \mathbf{x}) + \beta \cdot T(\mathbf{x}_{adv}, \mathbf{x})) \quad (11)$$

4.5 White Box Results

Here we measure the performance of our attacks in terms of ASR and CAD against their respective baselines over a range of compression rates.

Table 1 shows the success rates of the attacks with approximately equal distortion (\bar{D}). Although FAR (Shi et al., 2021) produces compressed adversarial

images, it fails to maintain attack effectiveness after the additional JPEG compression present in our pipeline. RCW results in a strong optimum, with superior success rates and minimal distortion levels. The attacks based on JPEG approximation and gradient projection perform well for stronger ϵ and thus higher distortion rates, but they fail to be effective for smaller distortion rates.

4.6 Black Box Results

Similar to the white box evaluations, we measure the performance of our attacks in terms of success rate and average distortion compared to their respective baselines over a range of compression rates. However, in these experiments, the target network is unknown. To simulate this scenario, we define the target model with a different architecture and weights than the surrogate model. For our experiments, the target model is InceptionV3 (Szegedy et al., 2016) pre-trained on ImageNet.

Table 2 shows the results of the attacks in a black box scenario. For smaller distortion rates, as required in this work, all attacks fail to fool the target model with different weights than the surrogate model $\hat{\phi}$. This is because black box attacks are a much more challenging problem than white box attacks, especially in combination with JPEG compression. FAR gives slightly better results than RCW, but with more than twice the distortion.

4.7 Hardened White Box Results

In the following, we present the results of our attack on models hardened by adversarial training. Adversarial training is currently the preferred way to make models more robust against adversarial attacks. We will compare two ResNets that were trained with the most prominent adversarial training protocols: PGD adversarial training (Madry et al., 2018) and FBF adversarial training (Wong et al., 2020).

Table 3 shows the results of the experiments performed on the hardened models. For the model that was trained with the FBF protocol, we see that all gradient projection attacks (Shi et al., 2021; Reich et al., 2024) struggle to maintain the success rate. RCW, which is based on adversarial optimization, manages to bypass the defenses and achieves high success rates at low distortion rates. Similarly, RCW achieves the best success rates for models hardened by the PGD adversarial training protocol. Although the samples were slightly more distorted than the FBF protocol experiments, they were still less distorted than any other related work we benchmarked against, with higher

Table 3: Conditional average distortion \bar{D} and attack success rate (ASR) per compression setting q in a scenario where the target model was trained using either PGD or FBF adversarial training.

Defense Models Experiments				
FBF				
Attack	CAD \bar{D}	ASR q=70	ASR q=80	ASR q=90
JpegBIM (Reich et al., 2024)	0.1450	0.078	0.088	0.089
JpegBIM (Shin and Song, 2017)	0.1450	0.078	0.088	0.089
FAR (Shi et al., 2021)	0.1435	0.013	0.013	0.026
RCW (ours)	0.1042	0.755	0.726	0.576
PGD				
Attack	CAD \bar{D}	ASR q=70	ASR q=80	ASR q=90
JpegBIM (Reich et al., 2024)	0.2901	0.053	0.065	0.055
JpegBIM (Shin and Song, 2017)	0.2853	0.050	0.057	0.058
FAR (Shi et al., 2021)	0.4786	0.007	0.008	0.015
RCW (ours)	0.2087	0.798	0.808	0.641

Table 4: This table shows the amount of perceived distortion of the adversarial samples. The success rates obtained were lower or equal to the those obtained by RCW. Lower values are better for both MAD and DISTS.

Perceived Distortion			
MAD			
Attack	70	80	90
JpegBIM (Reich et al., 2024)	0.6980	0.1890	0.1889
JpegBIM (Shin and Song, 2017)	0.6636	0.1752	0.1757
FAR (Shi et al., 2021)	66.4244	66.6663	67.6681
RCW (ours)	0.0015	0.0011	0.0006
DISTS			
Attack	70	80	90
JpegBIM (Reich et al., 2024)	0.0180	0.0117	0.0118
JpegBIM (Shin and Song, 2017)	0.0177	0.0115	0.0115
FAR (Shi et al., 2021)	0.1267	0.1070	0.1074
RCW (ours)	0.0015	0.0012	0.0009

success rates. To account for the fact that RCW has higher distortion rates in the case of the PDG Resnet, we adjust the settings of other methods to allow for higher distortion rates and thus higher success rates as well. For FAR (Shi et al., 2021) we use $\epsilon = 9e - 05$. Similarly, we increase ϵ to 0.0008 for the ensemble methods (Shin and Song, 2017; Reich et al., 2024).

4.8 Comparison of Perceived Distortion

Although L_p norms are still widely used to quantify the distortion in adversarial samples, many studies have found that they correlate poorly with human perception (Fezza et al., 2019). Therefore, an important quality to consider in adversarial samples is the amount of *perceived distortion*. This is the overall quality or fidelity of a sample as estimated by the human visual system. In this work, we use only L_p (\bar{D}) norms as a measure of *actual distortion* and

MAD/DISTS as a measure of *perceived distortion*. Note that we are testing for small distortion values, so all perceived distortion measures will be correspondingly small. Since adversarial samples are variable in distortion, we set the ASR as the baseline for comparison, with hyperparameters chosen so that the success rate is approximately equal to or less than the success rate of RCW in an appropriately small parameter grid in the white box scenario.

Table 4 shows the perceived distortion values obtained by the image quality evaluation methods. Although RCW always achieves a higher or equal success rate compared to the related work, its samples are much less distorted according to the perceived distortion metrics.

5 LIMITATIONS AND ETHICS

5.1 Analysis of the Compression Adaptation Search

Here, we analyze how well CAS approximates the true quality setting of the target system. We also motivate the search-based approach described in Section 3.2.1 by comparing it to a brute-force method that iterates over the entire set of possible compression intensities $\mathcal{Q} = \{1, \dots, 99\}$ to find the q with the smallest distance. Finally, we will perform an ablation study to isolate the effectiveness of both the JPEG approximation loss function extension and CAS in RCW.

5.1.1 Compression Estimation Analysis

For a target quality of 70, we run RCW on the test dataset and report the quality settings found by the search. We initialize the search with a budget of 150 steps and the temperature scalar τ , which progressively reduces the step size, set to 0.99. CAS returned the correct quality setting of 70 in every case. This ensures that using CAS instead of the aforementioned brute-force approach above will not have a negative impact on RCW’s attack success rate of RCW by inadvertently using an incorrect quality setting.

5.1.2 Benchmark Against Brute Force

A thorough comparison of CAS with the brute-force method outlined above requires an analysis of the performance differences in terms of the number of steps needed to reach an optimal q . As shown in Figure 6, CAS takes an average of 23 steps to reach q^* compared to a brute-force approach, which requires the

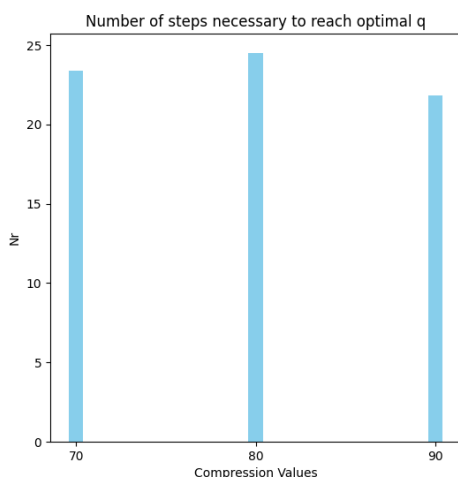


Figure 5: This chart shows the average number of steps required by CAS to reach q^* over a set of compression values in 10-increments.

processing of each quality setting and therefore takes 100 steps to reach the optimal q .

5.1.3 Ablation Study

To evaluate the benefit of the compression adaptation feature in RCW, we perform an ablation study by setting the compression value used for the gradient computation to a fixed value of $q = 80$ (as was done for other non-adaptive or non-ensemble methods such as FAR (Shi et al., 2021), see Section 4.2.). In our experiments, we will refer to this version of the attack as *approximate JPEG*. This attack optimizes similarly to RCW (see Equation 5), with the exception of $q = 80$.

$$\min_{\delta} \psi(\mathbf{x}, \mathbf{x}_{adv}, y_t, q = 80) \quad (12)$$

Finally, we include the original C&W attack by Carlini and Wagner (2017), which is the basis for RCW. This attack does not take compression into account. Table 5 shows the results of our ablation study. As shown, C&W (Carlini and Wagner, 2017) does not achieve acceptable success rates. As expected, including a JPEG approximation in the loss function with a fixed q results in high success rates for that particular q , but the model does not generalize to other quality settings. Not surprisingly, the less compression is used, the more effective C&W becomes. Finally, adding CAS results in RCW and in an attack that can successfully adapt to different compression rates.

5.2 Ethical Concerns

The study of adversarial attacks in machine learning presents both opportunities and ethical challenges. On the one hand, these attacks are invaluable for identifying weaknesses in models, allowing researchers to design systems that are more robust and secure. By understanding the ways in which models can be manipulated, researchers can develop defenses that prevent such exploits, ultimately making the use of machine learning more reliable, especially when it comes to high-security applications. However, the same research also raises significant ethical concerns, as the knowledge gained can be used for malicious purposes. Adversarial attacks can be used to deceive AI systems, bypass security measures, or even manipulate information. This can have harmful consequences. While adversarial research is essential for progress, it must be conducted with careful consideration of its potential for abuse. It must balance innovation with the responsibility to protect against malicious exploitation.

Table 5: Conditional average distortion \bar{D} and attack success rate (ASR) per compression setting q in a white box scenario. This ablation study compares C&W (Carlini and Wagner, 2017), a robust iterative attack with a fixed q for compression approximation, and RCW, which uses the JPEG approximation and CAS.

Ablation				
Attack	CAD \bar{D}	ASR q=70	ASR q=80	ASR q=90
C&W (Carlini and Wagner, 2017)	0.0665	0.061	0.109	0.221
+ Appr. JPEG	0.0684	0.131	0.664	0.115
+ CAS	0.1210	0.642	0.662	0.663

6 CONCLUSION & FUTURE WORK

Constrained adversarial optimization formulations provide an optimal basis for integrating differentiable JPEG approximations. However, using ensemble methods to account for different compression quality settings (Shin and Song, 2017) in target applications leads to long runtimes for attack methods that optimize to find a good balance between effectiveness and visual fidelity. We present a method that interrogates the target system once per sample and performs a compression adaptation search to find an optimal quality setting for the attack. Our approach allows us to compute adversarial samples that successfully defeat JPEG compression while maintaining high visual fidelity to the original sample. For nearly imperceptible amounts of distortion, our model outperforms the current state of the art in terms of success per perturbation in all experiments conducted, even overcoming a combination of compression and defensive strategies.

We now discuss possible future work. Replacing the gradient ensemble approach of existing methods Shin and Song (2017); Reich et al. (2024) with our compression adaptation search (CAS) suggests an advantage in terms of computational complexity, since we avoid the need for an additional inner loop in the optimization procedure (see Section 3.2.2). However, for future work, these advantages need to be investigated by conducting a performance benchmark that compares RCW to an adversarial optimization procedure that incorporates the established gradient ensemble method found in Shin and Song (2017) and Reich et al. (2024). Furthermore, although our attack can successfully bypass JPEG at different compression rates, there are other compression schemes that work differently internally. For example, JPEG2000 replaces the DCT with a wavelet transform to compute high frequency components (Taubman and Marcellin, 2002). Future work is needed to address these types of compression and have attacks successfully bypass them.

REFERENCES

- Akhtar, N., Mian, A., Kardan, N., and Shah, M. (2021). Advances in adversarial attacks and defenses in computer vision: A survey. *IEEE Access*, 9:155161–155196.
- Carlini, N. and Wagner, D. A. (2017). Towards evaluating the robustness of neural networks. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 39–57. IEEE Computer Society.
- Cozzolino, D., Poggi, G., Corvi, R., Nießner, M., and Verdoliva, L. (2023). Raising the bar of ai-generated image detection with CLIP. *CoRR*, abs/2312.00195.
- Ding, K., Ma, K., Wang, S., and Simoncelli, E. P. (2022). Image quality assessment: Unifying structure and texture similarity. *IEEE Trans. Pattern Anal. Mach. Intell.*, 44(5):2567–2581.
- Fezza, S. A., Bakhti, Y., Hamidouche, W., and Déforges, O. (2019). Perceptual evaluation of adversarial attacks for cnn-based image classification. In *11th International Conference on Quality of Multimedia Experience QoMEX 2019, Berlin, Germany, June 5-7, 2019*, pages 1–6. IEEE.
- Goodfellow, I. J., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 770–778. IEEE Computer Society.
- Kurakin, A., Goodfellow, I. J., and Bengio, S. (2017). Adversarial examples in the physical world. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Workshop Track Proceedings*. OpenReview.net.
- Kurakin, A., Goodfellow, I. J., Bengio, S., Dong, Y., Liao, F., Liang, M., Pang, T., Zhu, J., Hu, X., Xie, C., Wang, J., Zhang, Z., Ren, Z., Yuille, A. L., Huang, S., Zhao, Y., Zhao, Y., Han, Z., Long, J., Berdibekov, Y., Akiba, T., Tokui, S., and Abe, M. (2018). Adversarial attacks and defences competition. *CoRR*, abs/1804.00097.
- Liu, Z., Liu, Q., Liu, T., Xu, N., Lin, X., Wang, Y., and Wen, W. (2019). Feature distillation: Dnn-oriented JPEG compression against adversarial examples. In *IEEE*

- Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 860–868. Computer Vision Foundation / IEEE.
- Luo, C., Lin, Q., Xie, W., Wu, B., Xie, J., and Shen, L. (2022). Frequency-driven imperceptible adversarial attack on semantic similarity. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 15294–15303. IEEE.
- Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
- Reich, C., Debnath, B., Patel, D., and Chakradhar, S. (2024). Differentiable JPEG: the devil is in the details. In *IEEE/CVF Winter Conference on Applications of Computer Vision, WACV 2024, Waikoloa, HI, USA, January 3-8, 2024*, pages 4114–4123. IEEE.
- Shi, M., Li, S., Yin, Z., Zhang, X., and Qian, Z. (2021). On generating JPEG adversarial images. In *2021 IEEE International Conference on Multimedia and Expo, ICME 2021, Shenzhen, China, July 5-9, 2021*, pages 1–6. IEEE.
- Shin, R. and Song, D. (2017). JPEG-resistant adversarial images. In *NIPS 2017 workshop on machine learning and computer security*, volume 1.
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 2818–2826. IEEE Computer Society.
- Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I. J., and Fergus, R. (2014). Intriguing properties of neural networks. In Bengio, Y. and LeCun, Y., editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.
- Taubman, D. and Marcellin, M. (2002). Jpeg2000: standard for interactive imaging. *Proceedings of the IEEE*, 90(8):1336–1357.
- Tramèr, F., Kurakin, A., Papernot, N., Goodfellow, I. J., Boneh, D., and McDaniel, P. D. (2018). Ensemble adversarial training: Attacks and defenses. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
- Wang, X. and He, K. (2021). Enhancing the transferability of adversarial attacks through variance tuning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 1924–1933. Computer Vision Foundation / IEEE.
- Wong, E., Rice, L., and Kolter, J. Z. (2020). Fast is better than free: Revisiting adversarial training. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Zhao, Z., Liu, Z., and Larson, M. A. (2020). Towards large yet imperceptible adversarial image perturbations with perceptual color distance. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 1036–1045. Computer Vision Foundation / IEEE.